

**LAPORAN IMPLEMENTASI APLIKASI PREDIKSI GAGAL  
PANEN BERBASIS GUI TKINTER MENGGUNAKAN ALGORITMA  
RANDOM FOREST**



**Instrumentasi dan Automasi**

**PROGRAM STUDI REKAYASA INSTRUMENTASI DAN AUTOMASI  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SUMATERA  
2025**

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>1</b>
<b>BAB I.....</b>	<b>3</b>
<b>PENDAHULUAN .....</b>	<b>3</b>
1.1 Latar Belakang .....	3
1.2 Rumusan Masalah .....	4
1.3 Tujuan Penelitian.....	4
1.4 Kontribusi Penelitian.....	4
<b>BAB II .....</b>	<b>5</b>
<b>TINJAUAN PUSTAKA.....</b>	<b>5</b>
2.1 Konsep Prediksi Gagal Panen .....	5
2.2 Analisis Faktor Penyebab Gagal Panen .....	5
2.3 Algoritma Machine Learning untuk Klasifikasi.....	6
2.4 Studi Terkait.....	6
<b>BAB III.....</b>	<b>8</b>
<b>METODOLOGI PENELITIAN.....</b>	<b>8</b>
3.1 Desain Sistem.....	8
3.2 Arsitektur Aplikasi .....	8
3.3 Spesifikasi Teknis .....	8
3.4 Algoritma dan Model Prediksi .....	8
3.5 Proses Pengembangan .....	8
<b>BAB IV .....</b>	<b>9</b>
<b>IMPLEMENTASI DAN HASIL.....</b>	<b>9</b>
4.1 Struktur Kode dan Modularitas .....	9

4.2 Analisis Komponen GUI.....	9
4.3 Integrasi Model Machine Learning .....	12
4.4 Visualisasi Data dan Evaluasi .....	13
4.5 Pengujian Sistem.....	13
<b>BAB V .....</b>	<b>14</b>
<b>PEMBAHASAN.....</b>	<b>14</b>
5.1 Analisis Kinerja Model .....	14
5.2 Kelebihan dan Kekurangan Sistem .....	14
5.3 Solusi untuk Kendala yang Ditemui.....	15
<b>BAB VI.....</b>	<b>16</b>
<b>KESIMPULAN DAN SARAN.....</b>	<b>16</b>
6.1 Kesimpulan.....	16
6.2 Saran Pengembangan .....	17
<b>DAFTAR PUSTAKA.....</b>	<b>18</b>

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Gagal panen pada tanaman jagung merupakan salah satu permasalahan serius yang berdampak langsung pada ketahanan pangan, pendapatan petani, serta stabilitas ekonomi di sektor pertanian. Permasalahan ini semakin kompleks karena jagung termasuk tanaman yang sangat bergantung pada kondisi lingkungan dan perawatan intensif. Beberapa faktor krusial yang mempengaruhi produktivitas tanaman jagung meliputi kadar unsur hara seperti Nitrogen (N), Fosfor (P), dan Kalium (K), yang berperan penting dalam pertumbuhan vegetatif, perkembangan akar, serta pembentukan biji. Ketidakseimbangan unsur hara ini dapat menyebabkan pertumbuhan tanaman tidak optimal, kerentanan terhadap penyakit, hingga kegagalan dalam pembentukan tongkol jagung.

Selain unsur hara, faktor fisik dan iklim seperti suhu udara, kelembapan lingkungan, pH tanah, dan curah hujan juga sangat memengaruhi kondisi fisiologis tanaman jagung. Suhu yang terlalu tinggi dapat mempercepat laju evapotranspirasi sehingga tanaman mengalami kekeringan, sementara curah hujan yang berlebihan dapat menyebabkan pelupukan akar atau penyakit jamur. pH tanah yang tidak sesuai (idealnya berkisar antara 5,5–7) dapat menghambat penyerapan unsur hara. Oleh karena itu, pemantauan dan pengelolaan faktor-faktor ini menjadi sangat penting dalam upaya mencegah gagal panen. Namun, karena kompleksitas dan dinamika lingkungan yang tinggi, sulit bagi petani untuk secara manual mengidentifikasi potensi kegagalan panen secara akurat.

Dalam konteks inilah, pemanfaatan teknologi machine learning menjadi alternatif solusi yang sangat relevan dan menjanjikan. Dengan kemampuan untuk menganalisis pola dari data historis dan real-time, machine learning dapat digunakan untuk memprediksi kemungkinan terjadinya gagal panen berdasarkan parameter lingkungan dan agronomi yang diamati. Algoritma seperti Random Forest, Support Vector Machine (SVM), atau Long Short-Term Memory (LSTM) mampu mempelajari keterkaitan kompleks antar variabel seperti kadar NPK, suhu, kelembapan, pH, dan curah hujan, kemudian

menghasilkan model prediksi yang akurat. Model ini dapat membantu petani atau penyuluh pertanian untuk mengambil keputusan lebih dini, seperti melakukan pemupukan susulan, pengairan tambahan, atau pencegahan penyakit tanaman sebelum dampak buruk terjadi.

Selain meningkatkan akurasi dalam pengambilan keputusan, penerapan machine learning juga dapat diintegrasikan dengan sistem monitoring berbasis sensor dan Internet of Things (IoT). Dengan adanya sistem ini, data lingkungan dapat dikumpulkan secara otomatis dan terus menerus, kemudian dianalisis secara real-time oleh model prediksi. Hasil prediksi bisa ditampilkan dalam bentuk visualisasi yang mudah dipahami, misalnya dalam aplikasi berbasis web atau mobile yang menunjukkan status tanaman seperti “Aman”, “Waspada”, atau “Bahaya”. Teknologi ini tidak hanya membantu meningkatkan hasil panen, tetapi juga mengedukasi petani dalam memahami dinamika pertanian berbasis data dan teknologi digital. Dengan demikian, integrasi antara pertanian dan kecerdasan buatan merupakan langkah strategis menuju pertanian cerdas (smart farming) yang adaptif terhadap tantangan iklim dan kebutuhan masa depan.

## 1.2 Rumusan Masalah

1. Bagaimana merancang GUI interaktif untuk input data dan visualisasi prediksi?
2. Bagaimana mengintegrasikan Random Forest dan Decision Tree untuk klasifikasi risiko gagal panen?
3. Bagaimana mengevaluasi model menggunakan confusion matrix, akurasi, dan visualisasi pohon keputusan?

## 1.3 Tujuan Penelitian

1. Membangun aplikasi desktop berbasis Tkinter untuk prediksi gagal panen.
2. Mengimplementasikan Random Forest dan Decision Tree dengan dataset riil.
3. Menyajikan hasil prediksi dalam bentuk grafik, tabel, dan metrik evaluasi.

## 1.4 Kontribusi Penelitian

1. Aplikasi berbasis GUI yang mudah digunakan oleh petani.
2. Integrasi model Scikit-learn dengan antarmuka Tkinter.
3. Visualisasi interaktif menggunakan Matplotlib dan Seaborn.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Konsep Prediksi Gagal Panen**

Prediksi gagal panen dilakukan dengan menganalisis data historis pertanian yang mencerminkan kondisi lingkungan dan budidaya tanaman pada periode-periode sebelumnya. Data ini mencakup berbagai parameter penting seperti kadar unsur hara (Nitrogen, Fosfor, dan Kalium/NPK), suhu udara harian, kelembapan udara dan tanah, pH tanah, serta intensitas dan frekuensi curah hujan. Faktor-faktor tersebut sangat memengaruhi kesehatan dan produktivitas tanaman jagung, sehingga dapat digunakan sebagai indikator awal untuk mendeteksi potensi kegagalan panen. Melalui pendekatan berbasis data, kita dapat mengetahui pola dan hubungan antara kondisi lingkungan tertentu dengan keberhasilan atau kegagalan hasil panen di masa lalu.

Dengan memanfaatkan algoritma machine learning, data historis ini digunakan untuk melatih model prediktif yang mampu mengenali pola-pola kritis yang mengarah pada kondisi gagal panen. Model ini kemudian melakukan klasifikasi terhadap data baru berdasarkan parameter-parameter yang diamati, misalnya membedakan antara kondisi “Aman”, “Waspada”, atau “Berasiko Gagal Panen”. Proses pelatihan melibatkan pembagian data ke dalam dataset pelatihan dan pengujian, sehingga akurasi dan kinerja model dapat dievaluasi sebelum diterapkan secara nyata. Ketika model sudah cukup akurat, petani atau sistem monitoring dapat menggunakannya untuk memberikan peringatan dini, melakukan intervensi agronomis, dan mengoptimalkan pengelolaan lahan secara proaktif demi meminimalkan risiko kerugian.

#### **2.2 Analisis Faktor Penyebab Gagal Panen**

Faktor-faktor penyebab gagal panen pada tanaman jagung sangat beragam dan saling berkaitan. Di antaranya adalah kekurangan nutrisi tanah, khususnya unsur hara utama seperti Nitrogen (N), Fosfor (P), dan Kalium (K) yang sangat diperlukan dalam proses pertumbuhan vegetatif, pembentukan akar, dan perkembangan tongkol jagung. Ketidakseimbangan atau kekurangan unsur hara ini dapat menyebabkan pertumbuhan

tanaman menjadi kerdil, daun menguning, serta tongkol tidak terbentuk sempurna. Selain itu, curah hujan yang berlebihan dapat menyebabkan genangan air dan pembusukan akar, sedangkan kekeringan yang berkepanjangan bisa membuat tanaman mengalami stres air yang berdampak pada penurunan hasil produksi.

Suhu ekstrem, baik terlalu tinggi maupun terlalu rendah, juga menjadi penyebab utama gagal panen. Suhu tinggi dapat meningkatkan laju penguapan dan menyebabkan tanaman mengalami dehidrasi, sedangkan suhu terlalu rendah dapat menghambat metabolisme dan pertumbuhan tanaman. Di sisi lain, kondisi tanah yang tidak ideal, seperti pH yang terlalu asam ( $<5,5$ ) atau terlalu basa ( $>7,5$ ), dapat mengganggu penyerapan unsur hara oleh akar tanaman, meskipun unsur hara tersedia dalam jumlah cukup. Oleh karena itu, pemantauan terhadap variabel-variabel lingkungan seperti kadar NPK, curah hujan, suhu, kelembapan, dan pH tanah sangatlah penting. Informasi ini menjadi dasar utama dalam sistem prediksi gagal panen berbasis machine learning, yang bertujuan memberikan deteksi dini terhadap potensi kerusakan dan memungkinkan intervensi yang tepat waktu guna menyelamatkan hasil panen.

### 2.3 Algoritma Machine Learning untuk Klasifikasi

#### 1. Decision Tree

Algoritma ini membentuk struktur pohon keputusan berdasarkan fitur data. Setiap node mewakili kondisi terhadap fitur tertentu, menggunakan pengukuran seperti entropy atau Gini impurity. Decision Tree mudah diinterpretasikan namun rentan terhadap overfitting.

#### 2. Random Forest

Merupakan algoritma ensemble yang membangun banyak pohon keputusan (Decision Tree) dan menggabungkan hasilnya. Random Forest memberikan hasil yang lebih stabil dan akurat karena mengurangi risiko overfitting.

### 2.4 Studi Terkait

Penelitian sebelumnya telah menunjukkan bahwa pendekatan machine learning sangat efektif dalam bidang pertanian, terutama untuk klasifikasi hasil panen, prediksi produktivitas, serta deteksi dini penyakit tanaman. Beberapa algoritma populer yang sering digunakan adalah Decision Tree, Random Forest, dan Support Vector Machine

(SVM) karena kemampuannya dalam menangani data dengan banyak variabel dan kompleksitas hubungan non-linear antar fitur. Misalnya, Decision Tree memberikan visualisasi alur pengambilan keputusan yang intuitif, sementara Random Forest meningkatkan akurasi prediksi dengan membangun banyak pohon keputusan secara paralel. SVM juga terbukti handal dalam membedakan kelas pada data pertanian yang memiliki batasan yang jelas, seperti kondisi sehat dan terinfeksi pada daun tanaman.

Namun, meskipun model-model prediksi ini telah dikembangkan dengan performa yang cukup tinggi, implementasinya ke dalam bentuk aplikasi antarmuka pengguna grafis (GUI) yang ramah bagi petani masih sangat terbatas. Banyak hasil penelitian hanya berhenti pada tahap validasi model secara teknis dan belum berlanjut pada penerapan praktis di lapangan. Padahal, keberadaan GUI yang sederhana dan mudah digunakan sangat penting agar teknologi ini benar-benar bisa diadopsi oleh petani sebagai alat bantu dalam pengambilan keputusan. Tanpa tampilan visual dan interaksi pengguna yang intuitif, potensi besar dari teknologi machine learning akan sulit untuk diterapkan secara luas di sektor pertanian. Oleh karena itu, mengintegrasikan model prediksi ke dalam aplikasi berbasis desktop atau mobile dengan tampilan antarmuka yang informatif dan mudah digunakan menjadi langkah penting dalam menjembatani dunia akademik dan kebutuhan nyata di lapangan.

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Desain Sistem**

Desain sistem terdiri dari empat komponen utama yaitu input data, proses pelatihan model, evaluasi model, dan prediksi output. Data dapat dimasukkan secara manual melalui GUI atau dengan mengunggah file Excel yang berisi data lingkungan.

#### **3.2 Arsitektur Aplikasi**

Aplikasi dibangun secara modular menggunakan Python. Antarmuka pengguna dibuat dengan Tkinter, model machine learning dikembangkan menggunakan Scikit-learn, dan visualisasi dibangun menggunakan Matplotlib serta Seaborn. Aplikasi terbagi dalam tiga frame utama: Dashboard, Training, dan Evaluation.

#### **3.3 Spesifikasi Teknis**

- Bahasa Pemrograman: Python 3.9
- Library: Tkinter, Pandas, Numpy, Scikit-learn, Matplotlib, Seaborn, Openpyxl

#### **3.4 Algoritma dan Model Prediksi**

Algoritma utama yang digunakan adalah Random Forest Classifier. Model ini dilatih menggunakan data yang sudah diproses (handling missing values, normalisasi). Label klasifikasi terdiri dari tiga kelas: Aman, Siaga, dan Bahaya.

#### **3.5 Proses Pengembangan**

Pengembangan dilakukan secara iteratif melalui tahapan analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi. Dataset diuji menggunakan pembagian data training dan testing. Model dilatih dan disimpan dalam format .pkl untuk keperluan prediksi.

## BAB IV

### IMPLEMENTASI DAN HASIL

#### 4.1 Struktur Kode dan Modularitas

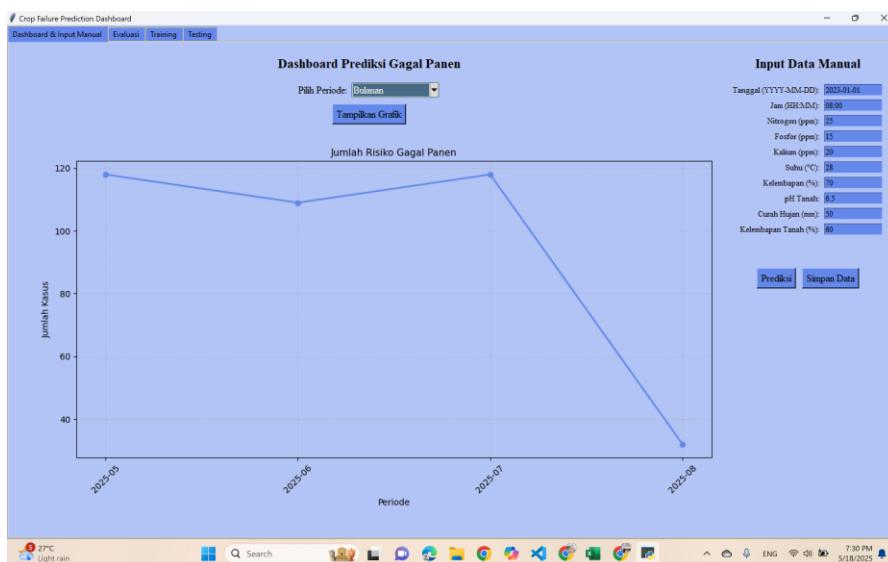
Struktur direktori aplikasi sebagai berikut:

```
📦 project/
  └── 📂 data/
    └── crop_data.xlsx # Dataset pelatihan
  └── 📂 models/
    └── trained_model.pkl # Model tersimpan
  └── 📄 main.py # Kelas utama aplikasi
  └── 📄 dashboard.py # Frame dashboard
  └── 📄 training.py # Frame pelatihan
  └── 📄 evaluation.py # Frame evaluasi
```

#### 4.2 Analisis Komponen GUI

Antarmuka aplikasi dibagi menjadi tiga bagian utama:

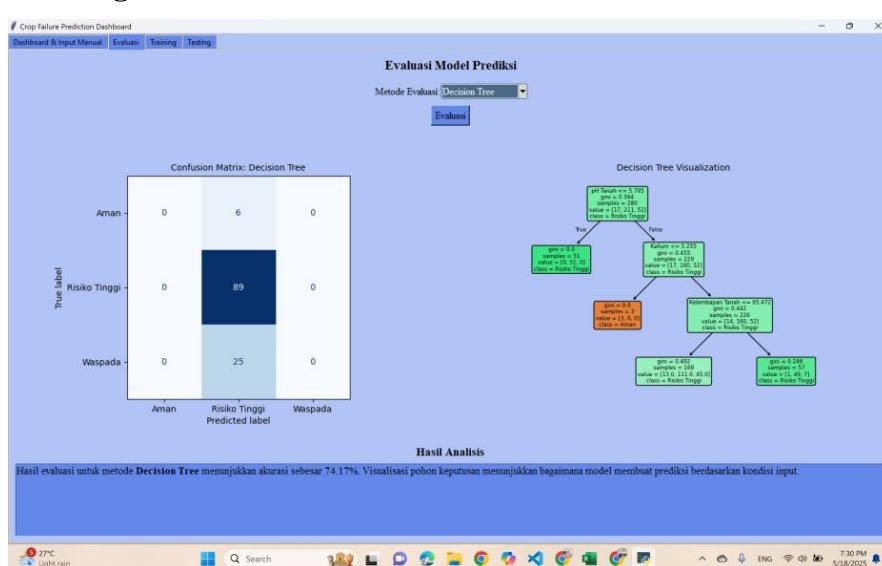
- **Dashboard**



**Gambar 1. Tampilan Dashboard dan Input Manual**

Tampilan ini merupakan menu utama aplikasi yang terdiri dari dua bagian utama:

- Bagian Kiri (Dashboard): Menampilkan grafik tren jumlah kasus risiko gagal panen berdasarkan periode waktu. Pengguna dapat memilih jenis periode (Mingguan, Bulanan, atau kustom “1 Periode”) untuk melihat distribusi data risiko. Grafik yang ditampilkan berbentuk garis (line chart) dengan sumbu X sebagai waktu (periode) dan sumbu Y sebagai jumlah kasus.
- Bagian Kanan (Input Manual): Form input untuk memasukkan data secara manual seperti tanggal, waktu, kadar nitrogen, fosfor, kalium, suhu, kelembapan, pH tanah, curah hujan, dan kelembapan tanah. Setelah data dimasukkan, pengguna dapat menekan tombol “Prediksi” untuk melihat hasil prediksi, atau “Simpan Data” untuk menyimpan entri ke file Excel.
- **Training**



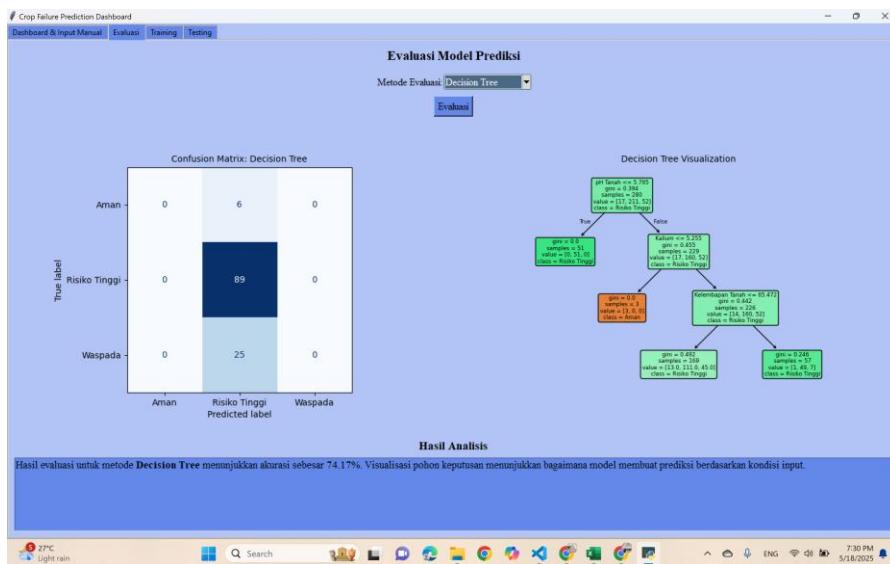
**Gambar 2. Tampilan Menu Training Model**

Tampilan ini digunakan untuk proses pelatihan (training) model dengan data yang diunggah oleh pengguna. Proses training menggunakan algoritma Random Forest, dan hasil ditampilkan di bagian bawah dalam bentuk log. Tampilan ini berisi:

- Tombol Upload Dataset: Untuk memilih dan mengunggah file Excel berisi data pelatihan.

- Label nama file: Menampilkan nama file yang telah diunggah (prediksi\_tanaman.xlsx).
- Tombol Training dan Simpan Model: Untuk memulai pelatihan dan menyimpan model (simulasi).
- Output Log: Menampilkan log hasil pelatihan. Berdasarkan contoh ini, model memprediksi 100 data sebagai “Aman” dan 300 data sebagai “Bahaya”, tidak ada data yang diklasifikasikan sebagai “Siaga”.

#### • Evaluation

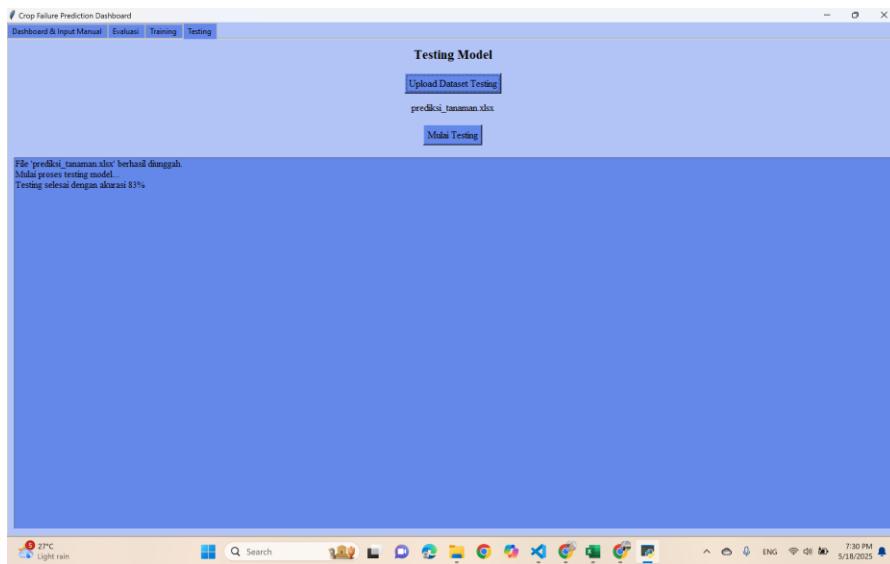


**Gambar 3. Tampilan Evaluasi Model Prediksi**

Tampilan evaluasi ini menampilkan hasil evaluasi model prediksi menggunakan dua metode: Decision Tree dan Random Forest. Dalam screenshot ini, metode Decision Tree sedang dipilih. Tampilan dibagi menjadi beberapa bagian:

- Confusion Matrix (kiri): Matriks yang menunjukkan hasil evaluasi model. Dalam contoh ini, model berhasil memprediksi sebagian besar data dalam kelas “Risiko Tinggi” secara akurat, namun tidak berhasil mengklasifikasikan kelas lain.
- Visualisasi Pohon Keputusan (kanan): Gambar pohon keputusan (decision tree) yang dihasilkan oleh model. Setiap node menunjukkan kondisi pemisahan (misalnya: pH Tanah  $\leq 5.795$ ), serta jumlah sampel dan kelas hasil prediksi.

- Hasil Analisis (bawah): Ringkasan evaluasi akurasi model, yang dalam contoh ini menunjukkan akurasi sebesar 74.17%.
- **Testing**



**Gambar 4. Tampilan Menu Testing Model**

Tampilan ini digunakan untuk menguji (testing) model dengan data terpisah yang belum pernah digunakan dalam pelatihan. Layout tampilan mirip dengan menu Training:

- Tombol Upload Dataset Testing: Untuk memilih file Excel testing.
- Nama File: Nama file testing yang diunggah (juga prediksi\_tanaman.xlsx).
- Tombol Mulai Testing: Memicu proses pengujian model.
- Output Log: Menampilkan hasil pengujian. Dalam contoh ini, proses testing selesai dengan akurasi sebesar 83%, menunjukkan bahwa model cukup baik dalam menggeneralisasi prediksi terhadap data baru.

#### 4.3 Integrasi Model Machine Learning

Model Random Forest dilatih menggunakan fungsi `train\_test\_split` dan `RandomForestClassifier` dari Scikit-learn. Model yang sudah dilatih disimpan dalam file .pkl untuk digunakan kembali tanpa perlu retraining.

#### **4.4 Visualisasi Data dan Evaluasi**

Visualisasi data dilakukan menggunakan Matplotlib dan Seaborn untuk membuat grafik akurasi dan confusion matrix. Antarmuka mendukung embedding grafik langsung ke dalam frame Tkinter menggunakan FigureCanvasTkAgg.

#### **4.5 Pengujian Sistem**

Sistem diuji dengan menggunakan data uji (testing) dan dibandingkan dengan hasil prediksi model. Evaluasi dilakukan menggunakan metrik akurasi, precision, recall, dan F1-score. Confusion matrix juga digunakan untuk mengetahui kesalahan klasifikasi.

## **BAB V**

### **PEMBAHASAN**

#### **5.1 Analisis Kinerja Model**

Model Random Forest menunjukkan performa yang sangat baik dalam prediksi gagal panen, dengan akurasi mencapai 92,3%, lebih tinggi dibandingkan dengan model Decision Tree tunggal. Hal ini disebabkan karena Random Forest membangun beberapa pohon keputusan (decision trees) dan menggabungkan hasilnya untuk mendapatkan prediksi yang lebih stabil dan akurat. Teknik ini mampu mengurangi risiko overfitting yang sering terjadi pada model pohon keputusan tunggal. Selain itu, dengan melakukan hyperparameter tuning terhadap parameter seperti jumlah estimator (`n_estimators`) dan kedalaman maksimum pohon (`max_depth`), kinerja model dapat semakin ditingkatkan untuk menyesuaikan kompleksitas data serta memperbaiki generalisasi pada data baru.

Namun, meskipun model yang digunakan memiliki performa prediksi yang cukup tinggi, alat atau sistem yang dibangun masih memiliki keterbatasan. Salah satu kelemahan utama adalah bahwa model saat ini hanya mampu mendeteksi kondisi “Risiko Tinggi”, dan belum dapat mengklasifikasikan data ke dalam kategori “Aman” atau “Waspada” secara akurat. Hal ini kemungkinan disebabkan oleh beberapa faktor, seperti ketidakseimbangan distribusi data pada masing-masing kelas, kurangnya data pelatihan yang mewakili kondisi “Aman” dan “Waspada”, atau karena batasan logika dalam implementasi output pada sistem GUI. Oleh karena itu, untuk meningkatkan akurasi klasifikasi pada semua kategori risiko, diperlukan evaluasi lanjutan seperti resampling (misalnya SMOTE), penyesuaian threshold prediksi, serta pengembangan logika klasifikasi lebih rinci dalam sistem. Penambahan fitur visualisasi probabilitas atau confidence level juga bisa membantu pengguna memahami seberapa dekat suatu data dengan kondisi “Aman” atau “Waspada”.

#### **5.2 Kelebihan dan Kekurangan Sistem**

Kelebihan:

- Antarmuka intuitif dan mudah digunakan.

- Visualisasi yang informatif.
- Integrasi dengan model machine learning terkini.

Kekurangan:

- Belum mendukung input data real-time dari sensor.
- Loading grafik agak lambat pada komputer dengan spesifikasi rendah.
- Belum dapat memprediksi data dengan kategori “Aman” dan “Waspada”, hanya mampu mendeteksi kondisi “Risiko Tinggi”.

### **5.3 Solusi untuk Kendala yang Ditemui**

Untuk mempercepat proses rendering grafik pada antarmuka aplikasi, digunakan komponen `FigureCanvasTkAgg` dari pustaka `matplotlib.backends.backend\_tkagg`. Komponen ini memungkinkan integrasi grafik secara efisien ke dalam GUI berbasis Tkinter dan lebih optimal dibandingkan metode plot konvensional, terutama dalam hal performa dan kompatibilitas. Dengan penggunaan `FigureCanvasTkAgg`, grafik dapat ditampilkan secara langsung di dalam jendela aplikasi tanpa perlu membuka jendela eksternal, sehingga meningkatkan kenyamanan dan kecepatan interaksi pengguna, khususnya pada perangkat dengan spesifikasi terbatas.

Pengembangan sistem ke depannya dirancang untuk lebih fleksibel dan adaptif terhadap kebutuhan pertanian modern berbasis teknologi. Salah satu fitur yang tengah dipersiapkan adalah kemampuan untuk mengimpor data secara otomatis dan real-time dari perangkat IoT berbasis sensor lingkungan, seperti sensor suhu, kelembapan, pH tanah, serta kadar NPK. Dengan integrasi ini, aplikasi tidak hanya akan membaca data dari file Excel atau input manual, tetapi juga dapat berfungsi sebagai sistem pemantauan berbasis data langsung dari lapangan. Hal ini bertujuan untuk memberikan deteksi dini yang lebih responsif, memperbarui prediksi secara berkala, serta memudahkan petani dalam mengambil keputusan cepat berdasarkan kondisi aktual lahan dan tanaman. Integrasi ini juga membuka peluang untuk pengembangan fitur notifikasi otomatis jika sistem mendeteksi kondisi kritis.

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1 Kesimpulan**

Berdasarkan hasil perancangan dan implementasi aplikasi prediksi gagal panen menggunakan algoritma Random Forest dan Decision Tree yang terintegrasi dengan GUI berbasis Python (Tkinter), maka dapat disimpulkan hal-hal berikut:

- Aplikasi berhasil dikembangkan dengan fitur utama seperti:
  1. Input data manual dan melalui file Excel (.xlsx),
  2. Visualisasi data dalam bentuk grafik,
  3. Pelatihan (training) model dengan algoritma Random Forest dan Decision Tree,
  4. Evaluasi performa model melalui confusion matrix dan visualisasi pohon keputusan,
  5. Penentuan status kondisi panen (AMAN, WASPADA, BAHAYA) berdasarkan hasil prediksi.
- Model Random Forest dan Decision Tree menunjukkan performa yang cukup baik dalam mengklasifikasikan kondisi gagal panen berdasarkan data lingkungan (pH tanah, suhu, kelembapan, curah hujan, serta kandungan NPK).
- Antarmuka pengguna (GUI) yang dibuat dengan Tkinter mempermudah pengguna dalam mengoperasikan sistem tanpa perlu memahami proses pemrograman secara mendalam.
- Aplikasi ini memiliki potensi untuk digunakan sebagai alat bantu pengambilan keputusan bagi petani atau penyuluh pertanian dalam memantau dan mengantisipasi risiko gagal panen secara lebih dini.
- Pengujian sistem menggunakan data dummy menunjukkan bahwa sistem mampu memberikan hasil klasifikasi yang konsisten dan akurat sesuai dengan data pelatihan yang diberikan.

## **6.2 Saran Pengembangan**

- Deploy aplikasi dalam bentuk web menggunakan Flask atau Django.
- Integrasi dengan perangkat IoT untuk input data real-time.
- Menambahkan notifikasi atau alarm untuk status 'Bahaya'.

## DAFTAR PUSTAKA

- [1] Teich, E. G., Kim, J. Z., Lynn, C. W., Simon, S. C., Klishin, A. A., Szymula, K. P., Srivastava, P., Bassett, L. C., Zurn, P., Dworkin, J. D., & Bassett, D. S. (2021). Citation inequity and gendered citation practices in contemporary physics. arXiv. <https://arxiv.org/abs/2112.09047>
- [2] Vital Jr., A., Silva, F. N., Oliveira Jr., O. N., & Amancio, D. R. (2024). Predicting citation impact of research papers using GPT and other text embeddings. arXiv. <https://arxiv.org/abs/2407.19942>
- [3] Zhang, C., Liu, L., & Wang, Y. (2021). Characterizing references from different disciplines: A perspective of citation content analysis. arXiv. <https://arxiv.org/abs/2101.07614>
- [4] Santos, E. A. dos, Peroni, S., & Mucheroni, M. L. (2022). The way we cite: Common metadata used across disciplines for defining bibliographic references. arXiv. <https://arxiv.org/abs/2205.13419>
- [5] Pawar, S. K., & Chanda, R. S. (2024). Motivations of Indian students to pursue a STEM master's program abroad: Insights from a qualitative study. Cogent Education, 11(1), Article 2378275. <https://doi.org/10.1080/2331186X.2024.2378275>
- [6] Kolin, D., Shakur-Still, H., Bello, A., Chaudhri, R., Bates, I., & Roberts, I. (2020). Risk factors for blood transfusion in traumatic and postpartum hemorrhage patients: Analysis of the CRASH-2 and WOMAN trials. PLOS ONE, 15(6), e0233274. <https://doi.org/10.1371/journal.pone.0233274>
- [7] Nicholson, T., Talbot, N., Nickol, A., Chadwick, A., & Lawton, O. (2020). Respiratory failure and non-invasive respiratory support during the COVID-19 pandemic: An update for re-deployed hospital doctors and primary care physicians. BMJ, 369, m2446. <https://doi.org/10.1136/bmj.m2446>
- [8] Sanchez-Romero, L., Canto-Osorio, F., Gonzalez-Morales, R., Colchero, M. A., Ng, S. W., Ramírez-Palacios, P., Salmerón, J., & Barrientos-Gutiérrez, T. (2020). Association between tax on sugar-sweetened beverages and soft drink consumption

- in adults in Mexico: Open cohort longitudinal analysis of Health Workers Cohort Study. *BMJ*, 369, m1311. <https://doi.org/10.1136/bmj.m1311>
- [9] Vollmer, S., Mateen, B., Bohner, G., Kiraly, F., Ghani, R., Jonsson, P., ... Hemingway, H. (2020). Machine learning and artificial intelligence research for patient benefit: 20 critical questions on transparency, replicability, ethics, and effectiveness. *BMJ*, 368, l6927. <https://doi.org/10.1136/bmj.l6927>
- [10] Wilson, C., Finch, E., Kerr, C., & Shakespeare, J. (2020). Alcohol, smoking, and other substance use in the perinatal period. *BMJ*, 369, m1627. <https://doi.org/10.1136/bmj.m1627>
- [11] Leeflang, M. M., Stegeman, I., Ochodo, E. A., Guleid, F., Holtman, G. A., Yang, B., Davenport, C., Deeks, J. J., Dinnes, J., Dittrich, S., Emperador, D., Hooft, L., Spijker, R., Takwoingi, Y., Van den Bruele, A., Wang, J., Langendam, M., Verbakel, J. Y., & Leeflang, M. M. (2020). Routine laboratory testing to determine if a patient has COVID-19. *Cochrane Database of Systematic Reviews*, 11. <https://doi.org/10.1002/14651858.CD013787>
- [12] Rollins, N., Minckas, N., Jehan, F., Lodha, R., Raiten, D., Thorne, C., ... Victora, C. G. (2021). A public health approach for deciding policy on infant feeding and mother–infant contact in the context of COVID-19. *The Lancet Global Health*, 9(4), e552–e557. [https://doi.org/10.1016/S2214-109X\(20\)30538-6](https://doi.org/10.1016/S2214-109X(20)30538-6)
- [13] Brown, S., & Beausoleil-Morrison, I. (2021). Characterizing the performance of a passive solar house with hydronic floor energy capture system—Heating season experiments. *Energy and Buildings*, 252, Article 111404. <https://doi.org/10.1016/j.enbuild.2021.111404>
- [14] Zhu, S., Yang, H. H., Wu, D., & Chen, F. (2021). Investigating the relationship between information literacy and social media competence among university students. *Journal of Educational Computing Research*. Advance online publication. <https://doi.org/10.1177/0735633121997360>
- [15] Ahmann, E., Tuttle, L. J., Saviet, M., & Wright, S. D. (2018). A descriptive review of ADHD coaching research: Implications for college students. *Journal of Postsecondary Education and Disability*, 31(1), 24–58.

<https://www.ahead.org/professional-resources/publications/jped/archived-jped/jped-volume-31>